APPENDIX A

FOIA-2005-0505

Best Available Copy

PO8150 05/15/50

Additional Information for Example Embodiments of Dynamic Ad Insertion

The Activate ad insertion service, in one embodiment, utilizes broadcast content that has been prepared to be transmitted in a digital encoded manner, such as that provided by a digital encoder that produces media files for distribution over the Internet. Preferably, the content to be broadcast (for example, a radio broadcast) has inserted within it event markers that signal the beginning of an advertisement and an ending of an advertisement. Then, when the broadcaster (or other content supplier) supplies the media clip to the Activate Network, it is supplied with digital encodings (the event markers) that signal the locations within the content where the dynamic ads may appear. As mentioned, ads can appear anywhere within the content. The ASX file capabilities are then used to cause the media player to perform specific actions when these event markers are detected. For example, in one embodiment, the media player issues an "Ad URL Request" to the 3rd party Ad Server in response to detecting an Begin_Ad event marker. This request, as shown in other diagrams, results in the media player playing a specific ad as indicated by the 3rd Party Ad Server. The media player then resumes playing standard content when it detects an "End_Ad" event marker.

Activate.net
Live Radio Ad Insertion Requirements

Overview:
There are three possibilities for a live radio ad insertion offering: 1) Open system: It is called open because the ad sits outside the radio station and is dynamically called up from an ad serving company. 1a) Open system which works with AdForce, 1b) Open system which works with Engage and allows for ad targeting/profiling. 2) Closed system: It is called closed because the ad sits within the closed environment of the radio station. The station sells the ads and holds its own ad inventory.

General Requirements for all options:

Format: Prefer support for WMT immediately, followed by Real and of QT (when we roll this out). Potential support in future needed for MP3
Data rates: for WMT & Real up to 28.8 (MP3 would most likely be at a higher data rate)
In stream ads: Pre roll and interstitial
Reporting: Ability to associate the ad to the original content requested, ability to determine who listened to **complete** ad.
Support: Must be able to provide support and assistance with entire ad insertion process

Product development and roll out: Roll out one automation system first. A manual system is really a back up option for stations that either have different automation systems or no automation system at all.

1a) OPEN SYSTEM: Works with AdForce

1. Automation/Scalability: Must be an automated, scalable solution to handle millions of ad calls
2. Redirects: Must be able to handle "redirect" ad calls via industry ad serving firms like AdForce and/or Doubleclick.
3. Ad Provisioning: Provisioning of ad "avails" and campaigns should be done via existing industry interfaces that publishers and advertisers currently use for banner ads.
4. Ad scheduling: Both local (station) or remote (network) control of spot scheduling & content is ideal. Network control required - local control optional.
5. End user experience: Transition from content to ad and back to content must be seamless to the human eye and ear. No black space or dead air.
6. End user experience: Smooth ad insertion - no "chopping" or airing parts of on-air spots, matched levels & audio density (i.e.: if station audio is heavily processed, spots should be as well).

7. Station integration: Easy interface to station automation system (or - optionally - control board if station still uses carts for spots). That is, the vendor's solution should be able to interface easily with the existing automation system at the radio station, with little configuration or additional software for station personnel to install.
8. Station integration: Ideal situation would require no change to audio of spots used on-air (no subaudible tones or watermarking).
9. Station integration: Ideally all station-side software should run on existing encoder - no extra box just for inserts.

1b) OPEN SYSTEM: Works with AdForce & Engage

1-9 in above with the addition of :
10. System needs to be capable of supporting Engage profile ad targeting.

CLOSED SYSTEM: Station provisions and sells ad

1. Automation/Scalability: Must be an automated, scalable solution to handle tens of thousands of ads per month
2. Ad scheduling: Potentially provide enabling tool to allows stations to book ads, and/or interface with the stations traffic manager
3. End user experience: Transition from content to ad and back to content must be seamless to the human eye and ear. No black space or dead air.
4. End user experience: Smooth ad insertion - no "chopping" or airing parts of on-air spots, matched levels & audio density (i.e.: if station audio is heavily processed, spots should be as well).
5. Station integration: Easy interface to station automation system (or - optionally - control board if station still uses carts for spots). That is, the vendor's solution should be able to interface easily with the existing automation system at the radio station, with little configuration or additional software for station personnel to install.
6. Station integration: Ideal situation would require no change to audio of spots used on-air (no subaudible tones or watermarking).
7. Station integration: Ideally all station-side software should run on existing encoder - no extra box just for inserts.

Station automations systems to consider:
Broadcast Electronics (AudioVault):
RCS (Selector)
Prophet Systems (Wizard)
Scott Studios

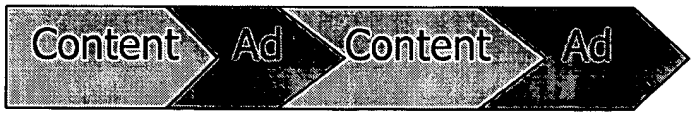
Potential Beta customers
Broadcast Programming
Entercom
Fisher Broadcasting

Activate Dynamic Ad Insertion for On Demand Content

Product Description

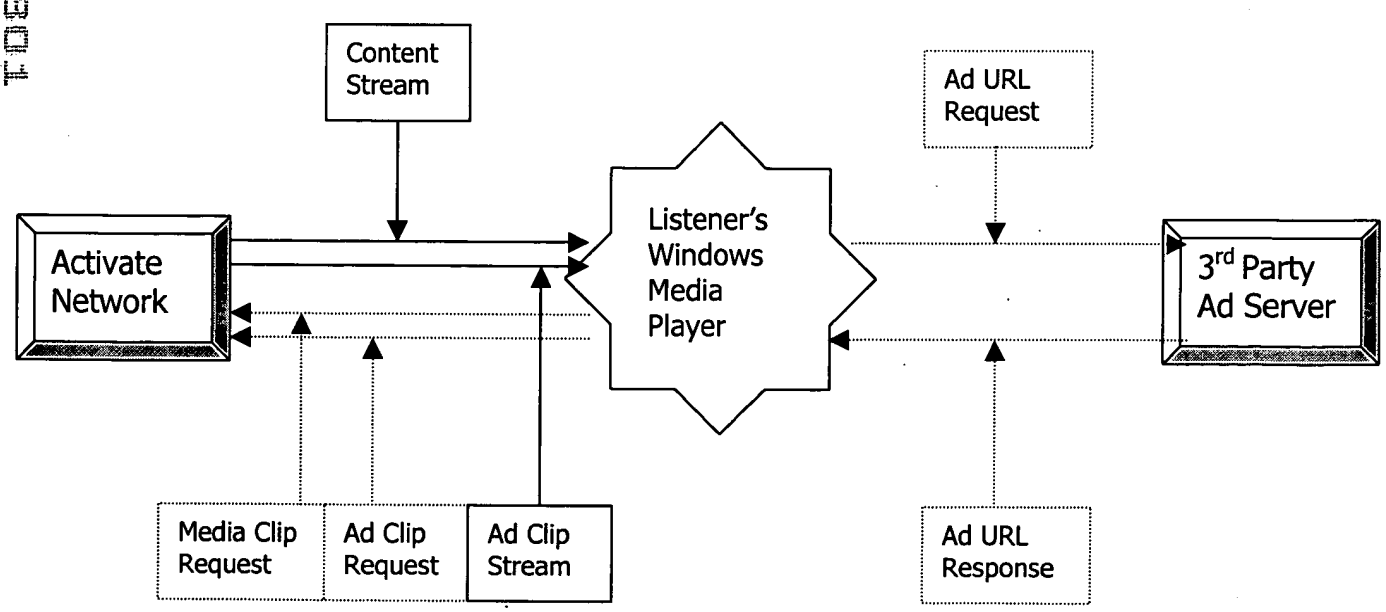
The Activate ad insertion service involves placing ads within the content stream of Windows Media player files. This is done dynamically by using existing technology that is part of WMT's ASX file capabilities along with leveraging the AdForce and Engage 3rd party ad serving networks.

Spot availabilities are created within the encoded Streaming Media file so that ads can be inserted in a fashion similar to the model used for broadcast television and radio.



Ads can appear at the beginning, anywhere within the content and at the end of the content file if desired. Activate will be providing templates and a developer tool kit for content owners who wish to create these kinds of spot availabilities in their content on their own or will provide this service for content owners on a fee basis.

What's new about this offering is that in stream ads can now by dynamically served similar to how banner ads are served today. This means that an http call is actually being made from the content stream to a 3rd party ad server like AdForce or Engage, and they are deciding which ad needs to be streamed for each viewer. Once the AdForce or Engage server has identified the correct ad to stream, it tells the player to get the ad file from Activate's network, which in turn streams it back to the player. The Widows Media player then starts to buffer the incoming ad stream while the content continues streaming on the player. When the time arrives to play the ad, Windows Media player switches the ad stream and plays it in its entirety. Once the ad has been played the player then resumes playing the content.



Activate Proprietary & Confidential Information Subject to NDA

Length and data rates of ads are as follows.

Audio/Video	Length in Seconds	Data Rates
Audio	10, 15, 30. 60	22 kbps
Video	10, 15, 30. 60	45 kbps
Video	10, 15, 30. 60	100 kbps

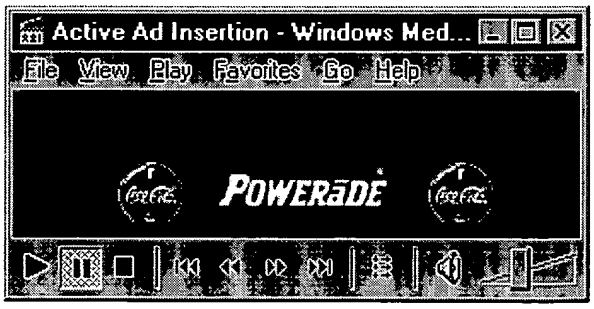
Just like Banner ads, these Streaming Media ads can be targeted since, they are being dynamically served by the AdForce and Engage 3rd party ad serving platforms. Targeting can be as simple as utilizing the individual's IP address or as sophisticated as the Engage profiling method.

The existing AdForce Desktop 3.1 and Engage Ad Manager provisioning and campaign systems are being modified to accept this new streaming media ad format. This means that content owners will now be able to utilize these tools in a similar fashion to how web publishers do for placing banner ad locations on their web pages.

Advertisers and Ad Agencies will also be able to use the established AdForce and Engage Banner ad placement tools to run streaming media ad campaigns similar to how they do for banner ads today. By leveraging these existing web based 3rd party ad serving systems, publishers and advertisers will also have access to reporting information comparable to what they have now with Banner ads.

Banner ads can also be displayed within the Windows Media Player and provide interactivity similar to existing banner ads that allow viewers click on the banner to obtain more information about the advertised product or service

Media Player for Audio Ad with Banner Ad



Activate Proprietary & Confidential Information Subject to NDA

Click Through from Banner Ad to Advertiser's Web Page

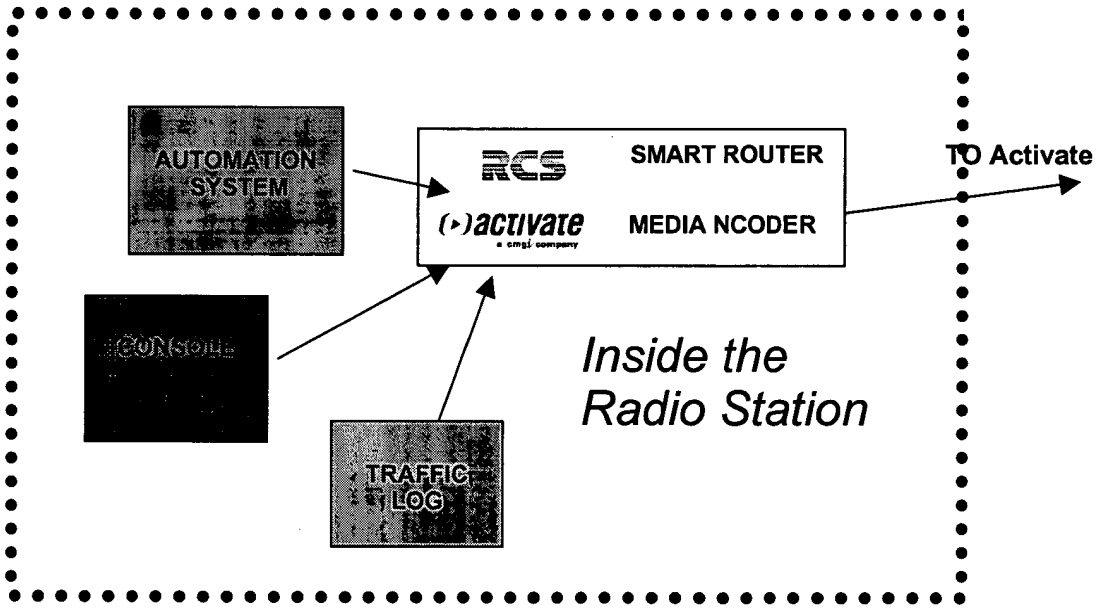
EDS1500-0055705

Activate Dynamic Audio Ad Insertion for Live Content

Product Description

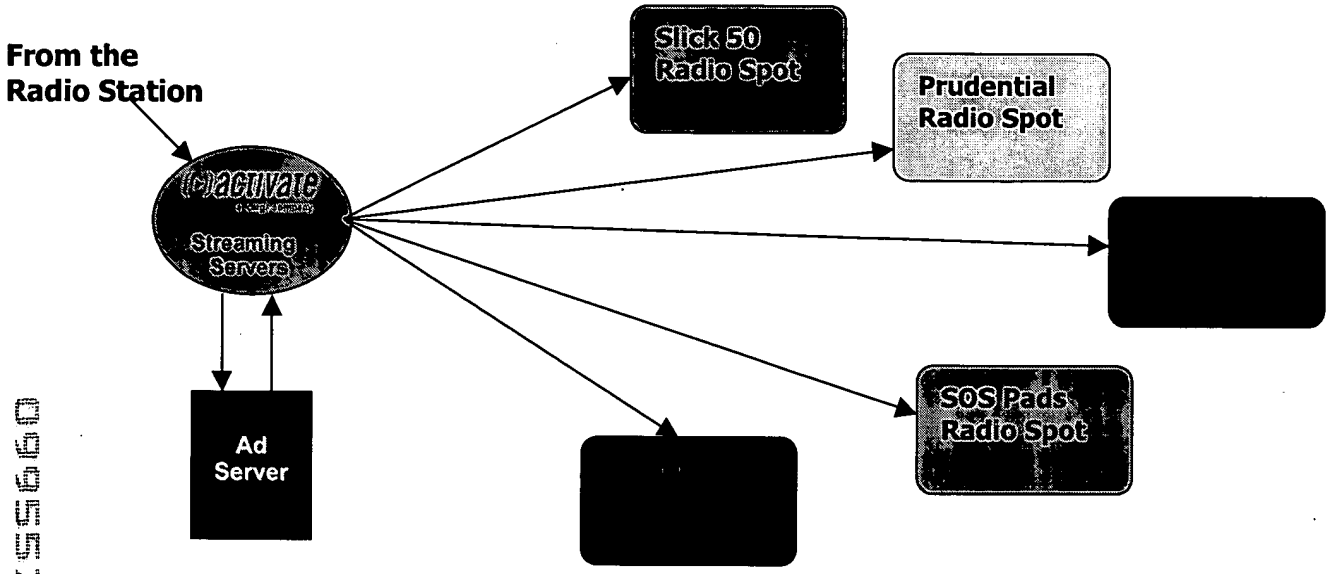
Similar to the previously described On Demand Dynamic Ad Insertion service, the Live version also places ads within the content stream of Windows Media player files. This is done dynamically using the same WMT ASX file technology and 3rd party ad serving platforms in conjunction with an additional piece of software.

The live audio stream first enters an Activate Windows Media Encoder box where the insertion software listens in real time to the pre-encoded audio stream in order to determine when the next ad break will occur and the length of the upcoming ad spots. Once it has identified these upcoming spot availabilities, it creates matching Windows Media events on the fly, which in turn request ads from either AdForce or Engage 3rd party ad servers. The insertion software relies on either the code signals sent by the progaming/traffic automation system or the contact closure signals that are generated by audio mixing console boards whenever a channel is opened or closed.



Activate Proprietary & Confidential Information Subject to NDA

This live ad insertion capability has been designed primarily for Terrestrial and Internet radio stations that want to strip out the original "broadcast" ad from their audio stream and replace it with a dynamically served Internet spot. Just like On Demand Streaming Media ads these ads can also be targeted since, they are being dynamically served by the AdForce and Engage 3rd party ad serving platforms. Targeting can be as simple as utilizing the individual's IP address or as sophisticated as the Engage profiling method.



HIGHLY CONFIDENTIAL

General Requirements for Live Audio Ad Insertion

Format: Prefer support for WMT immediately, followed by Real and of QT (when we roll this out). Potential support in future needed for MP3

Data rates: for WMT & Real up to 28.8 (MP3 would most likely be at a higher data rate)

In stream ads: Pre roll and interstitial

Reporting: Ability to associate the ad to the original content requested, ability to determine who listened to **complete** ad.

Support: Must be able to provide support and assistance with entire ad insertion process

Automation/Scalability: Must be an automated, scalable solution to handle millions of ad calls

Redirects: Must be able to handle "redirect" ad calls via industry ad serving firms like AdForce and Engage.

Ad Provisioning: Provisioning of ad "avails" and campaigns should be done via existing industry interfaces that publishers and advertisers currently use for banner ads.

Ad scheduling: Both local (station) or remote (network) control of spot scheduling & content is ideal. Network control required - local control optional.

End user experience: Transition from content to ad and back to content must be seamless to the human eye and ear. No black space or dead air.

End user experience: Smooth ad insertion - no "chopping" or airing parts of on-air spots, matched levels & audio density (i.e.: if station audio is heavily processed, spots should be as well).

Station integration: Easy interface to station automation system (or - optionally - control board if station still uses carts for spots). The solution should be able to interface easily with the existing automation system at the radio station, with little configuration or additional software for station personnel to install.

Station integration: Does not require any changes to audio of spots used on-air (no subaudible tones or watermarking).

Station integration: Software should run on existing encoder - no extra box just for inserts.

Targeting: System needs to be capable of supporting Engage profile ad targeting.

ACTIVATE-005756

Dynamic Ad Insertion

System Requirements Document

Version 0.1
September 8, 2000

This document contains trade secrets and proprietary information of ACTIVATE.NET. No use or disclosure of the information contained herein is permitted without prior written consent.

Activate.net
1130 Rainier Avenue, South
Seattle, WA 98144
(206) 830-5300
http://www.activate.com

1. Overview

This document defines some of the requirements for the two components of Activate's dynamic ad insertion offering: i) the system for ad insertion in On Demand content streams; and ii) the system for ad insertion in Live content streams. The requirements for the On Demand system integrate the technology capabilities of Activate's development partner companies AdForce and Engage for third-party ad serving networks. The requirements for the Live system incorporate the technology capabilities of Activate's development partner companies RCS (for software that monitors live radio streams) and Engage (for third-party ad serving and profiling).

The objective of this document is to establish the criteria for the development and implementation of Activate's dynamic ad insertion systems.

3. On Demand Content

Activate's dynamic ad insertion system for On Demand content will insert advertisements within audio and video content streams on the Internet, resulting in a user experience that is similar to broadcast radio and television. Activate's system is based on existing Microsoft Windows Media Technology as well as the technology of third-party ad serving networks from companies AdForce (and Engage).

3.1 General

The mandatories for Activate's ad insertion system for On Demand content are to:

1) Provide an automated, scalable solution that can handle millions of ad calls.
2) Deliver an end-user experience that is seamless to the human eye and ear – transitions from content to ad and back to content must be smooth and without black space on screen or dead air in audio.
3) Deliver reliable and consistent ad insertions, both audio and visual – no "chopping" or airing portions of on-air spots, and matching audio levels and density.
4) Provide robust reporting of ad-content associations and listener metrics.
5) Provide customer support and assistance throughout entire ad insertion process.

3.2 Technology

1) Format: Initially, must support Windows Media Technology and Windows Media Player, version 6.4 and higher. Future support for Real and QuickTime formats.
2) Operating System: Must run on Microsoft Windows 95/98/2000 and Microsoft Windows NT 4.0 and higher.
3) Memory Resources: Must use no more than *xxxxxx (TBD)* memory resources. PC's with minimum 64MB RAM.
4) Browser: Must run on Microsoft Internet Explorer 4.0 and higher.
5) Data Rates: Must function efficiently on systems with 28.8k modems and higher. Target users with 56k and higher connection speeds.

6) Performance Window: *(TBD)*

3.3 Production and Broadcast Operations

1) Content Acquisition

- Media Content Types
  a) Encoded
  b) Not encoded

- Ad Content Types
  a) Encoded - With or without banner?
  b) Not encoded
  c) Ad spot durations = 30 or 60-seconds

2) Work Order

- Web-based
- Linked to ScheduAll
- Automated email confirmation and results

3) Production Standards

Data Rates and Aspect Ratio (Image Window)

Format	Data Rate	Frame Size
Audio	22 kbps	176x132 (w/GIF)
Video (target 56k)	37 kbps	176x132
Video (target 100k)	100 kbps	192x144

3.4 Customer Care

1) Configuration
- Tier1, Tier2, HelpDesk; web-based

2) Ownership Scenarios
- Activate customer
- AdForce customer

3) Support Elements
- Ordering
- Product Questions
- Billing

- NOC Status

4. Live Content

Activate's dynamic ad insertion system for Live content will insert advertisements within audio content streams on the Internet. The initial target market for this system consists of radio stations, and therefore the system design requirements reflect the technology capabilities and operating processes of that customer segment. The ad insertion system for Live content incorporates the specifications of RCS's SplitStream software that monitors live radio streams, and Engage's AdManager ad serving and profiling system.

4.1 General

The mandatories for Activate's ad insertion system for Live content are to:

1) Provide an automated, scalable solution that can handle millions of ad calls.
2) Deliver an end-user experience that is seamless to the human eye and ear – transitions from content to ad and back to content must be smooth and without black space on screen or dead air in audio.
3) Deliver reliable and consistent ad insertions, both audio and visual – no "chopping" or airing portions of on-air spots, and matching audio levels and density.
4) Provide an easily installable and maintainable interface to radio station's existing automation system (or control board if appropriate) and hardware.
5) Provide both pre-roll and interstitial in-stream ads.
6) Provide robust reporting of ad-content associations and listener metrics.
7) Provide customer support and assistance throughout entire ad insertion process.
8) Integrate technology to manage "redirect" ad calls via Engage's system.
9) Integrate technology that provisions ad "avails" and ad campaigns via existing industry interfaces that publishers and advertisers currently use for banner ads.
10) Provide capability for remote (network) control of ad spot scheduling and content; local (radio station) control of ad spot scheduling and content is optional.
11) Deliver a service that supports Engage's profile ad targeting functionality.

4.2 Technology

1) Format: Initially, must support Windows Media Technology and Windows Media Player, version 6.4 and higher. Future support for Real and QuickTime formats.
2) Operating System: Must run on Microsoft Windows 95/98/2000 and Microsoft Windows NT 4.0 and higher.
3) Memory Resources: Must use no more than *xxxxxx (TBD)* memory resources. PC's with minimum 64MB RAM.
4) Browser: Must run on Microsoft Internet Explorer 4.0 and higher.
5) Data Rates: Must function efficiently on systems with 28.8k modems and higher. Target users with 56k and higher connection speeds.
6) Buffering/Latency: Maximum 10-second latency between ad and content.
7) Performance Window: Weekdays (Monday-Friday) during peak listening hours (7:00 a.m. – 4:00 p.m.). Weekends and holidays are not mandatory, initially.
8) Player: No proprietary player.
9) Platform Detection: Must be able to sniff WindowsNT platform.

HIGHLY CONFIDENTIAL

10) Ad Calls: Preferably uses Microsoft WMT ASX "Event" and "Open Event" calls.

4.3 Production and Broadcast Operations

1) Content Acquisition

- Media Content Types
  a) Encoded at radio station

- Ad Content Types
  d) With or without banner
  e) Ad spot durations = 15, 30, or 60-seconds

2) Work Order

- Web-based
- Linked to ScheduAll
- Automated email confirmation and results

3) Production Standards

Data Rates and Aspect Ratio (Image Window)

Format	Data Rate	Frame Size
Audio	22kbps	176x132 (w/GIF)

4.4 Customer Care

4) Configuration
- Tier1, Tier2, HelpDesk; web-based

5) Ownership Scenarios
- Activate customer
- Engage customer
- RCS customer

6) Support Elements
- Ordering
- Product Questions
- Billing
- NOC Status

7) Escalation Process

4.5 Ordering

1) Method
- Manual, initially.
- Online

2) Elements

3) Timing
- Windows

4) Initiator
- Contact – sales person, producer, etc.?

4.6 Billing

1) Method
- Manual, initially.
- Online

FOIA 05-0559

T08T60" 05755660

Ad Insertion and Now Playing Partnership Presentation

Prepared for:
Feed The Monster Media, Inc.

June 29, 2000

activate
a cmgi company

RCS

Proprietary & Confidential

Agenda

10:30-11:00 Overview of Partners

11:00-12:00 Ad insertion

12:00-12:30 Radio Show

Activate Overview

- Turn-key streaming solutions to radio industry
- Robust, scalable network
- Streaming 15 of top 25 Arbitron Infostream rated stations
- Hands-on experience with large-scale roll-outs

Proprietary & Confidential

activate a cmgi company RCS

RCS ...we do RADIO

World's leading provider of broadcast software

- Twenty years of broadcast experience
- International presence
- Selector® used by 85% of music radio stations in America
- 24-hour tech support

RadioShow (▸)activate a cmgi company RCS

Proprietary & Confidential

AdForce/Engage Overview

- **Largest independent ad serving & management solution**
- **Serving over 16 billion ad impressions/mo**
- **End-to-end ad management**
  - Scheduling
  - Tracking
  - Reporting
- **Profile based targeting**
- **Relationships with over 2,000 ad agencies and advertisers**

(▸)activate a cmgi company RCS

Partnership Ad Insertion Solution

- **Server side ad insertion**
- **End-to-end management system**
  - Scheduling, targeting, serving, reporting
  - Ad agency and advertiser friendly
- **Proven, experienced team**
  - Integration vs. invention
  - All are service companies 7x24x365
  - Well-financed to grow into future
- **Minimal disruption to station activities**

Proprietary & Confidential

RadioShow from RCS

- It's an Internet audio appliance with…
  - Interactive Advertising
  - Targeted Ad Insertion
  - The "Buy Me Button"™
  - Dynamic animated graphics

activate a cmgi company RCS

Why RadioShow?

- New revenue stream
- Enforces station branding
- Makes site more sticky
- Adds motion & life to the site

activate a cmgi company RCS

RadioShow shows…

- What's playing now?
- Metadata for song
- Interactive advertising
- Animated graphics
- Animated ads
- "Buy Me Button"™

…and it **increases TSLL**

activate a cmgi company RCS

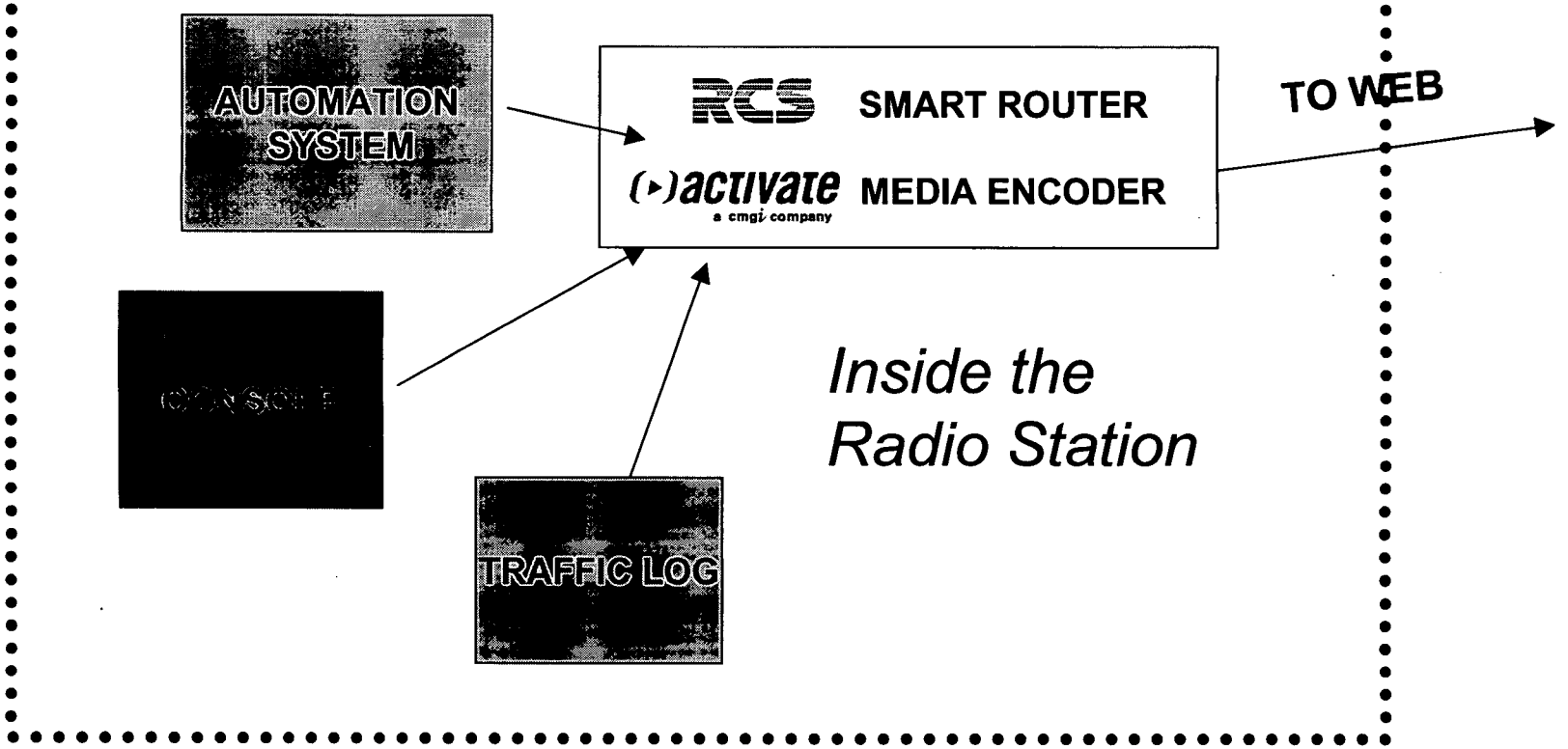
What is TSLL?

Time
Spent
Looking &
Listening

activate a cmgi company RCS

Dynamic Audio Player & Smart Spot Splitter

RadioShow



activate a cmgi company RCS

Dynamic Audio Player & Smart Spot Splitter

RadioShow

From STATION

activate a cmgi company Streaming Servers

Branded Player for KJLS-FM

The LLs (Lookers/Listeners)

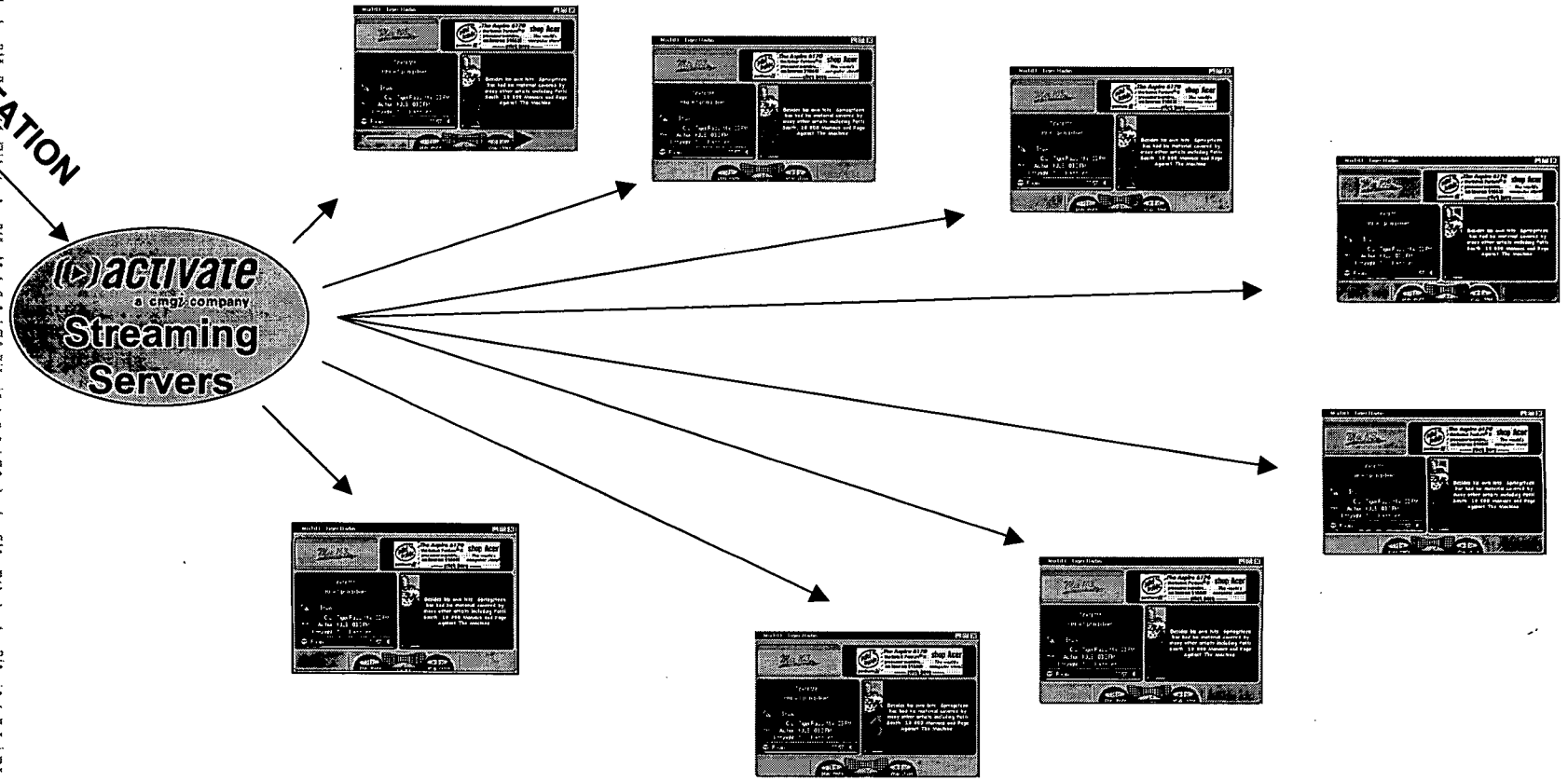
activate a cmgi company RCS

Proprietary & Confidential

...Same program content to all, with continuously changing graphics

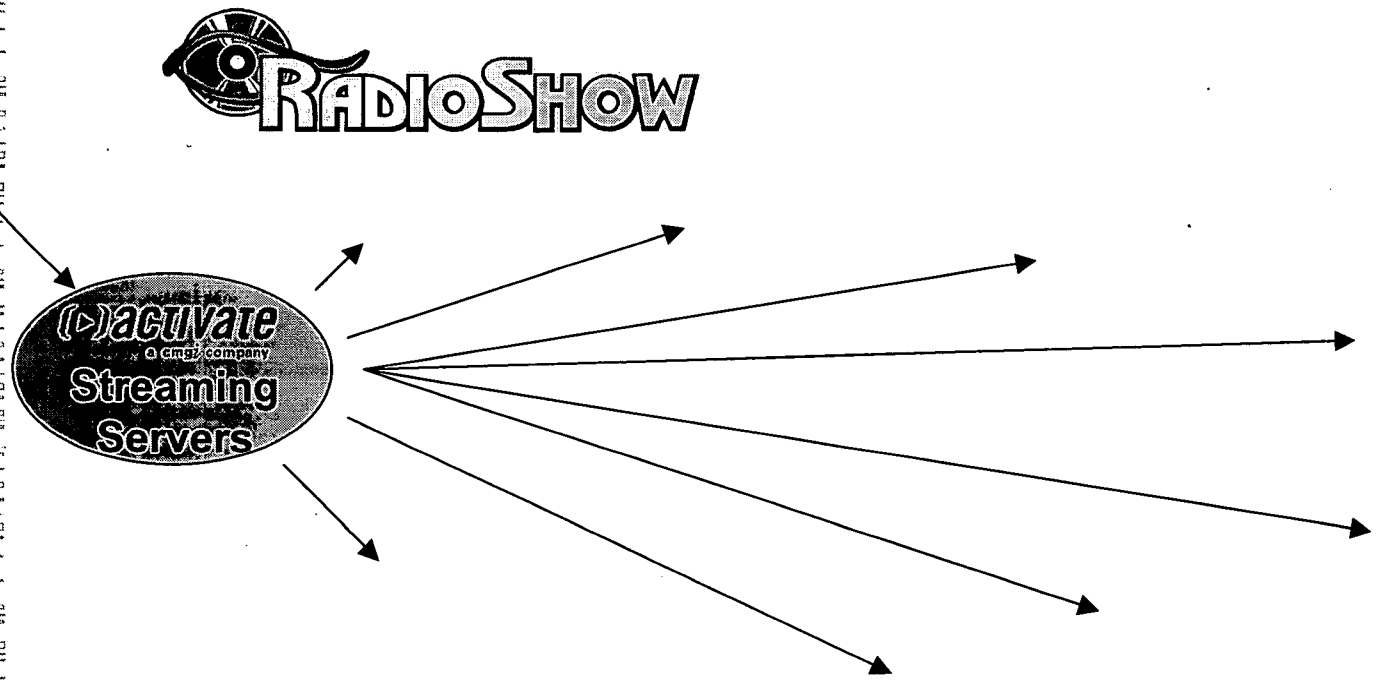
RadioShow

From STATION

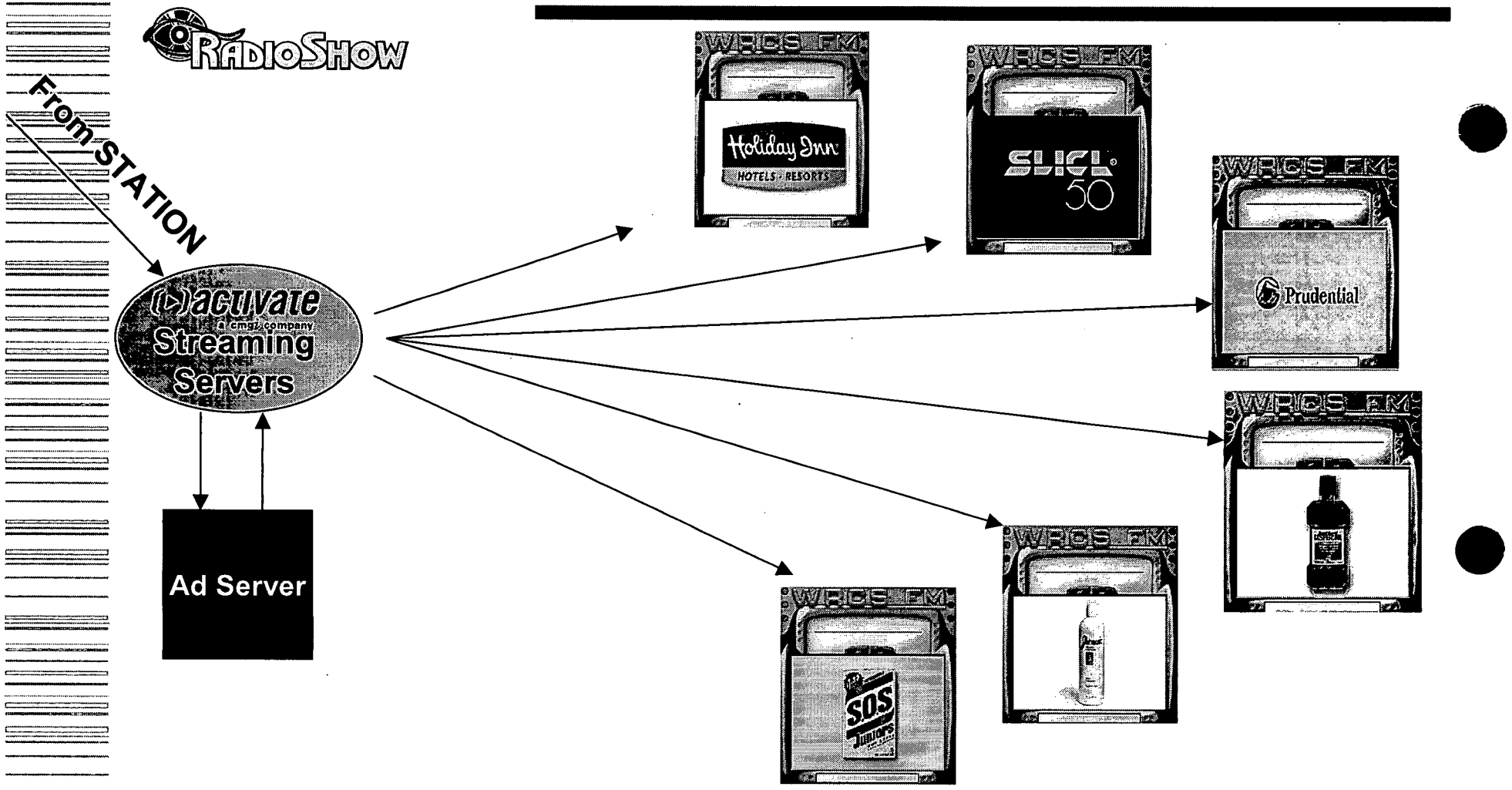


activate a cmgi company RCS

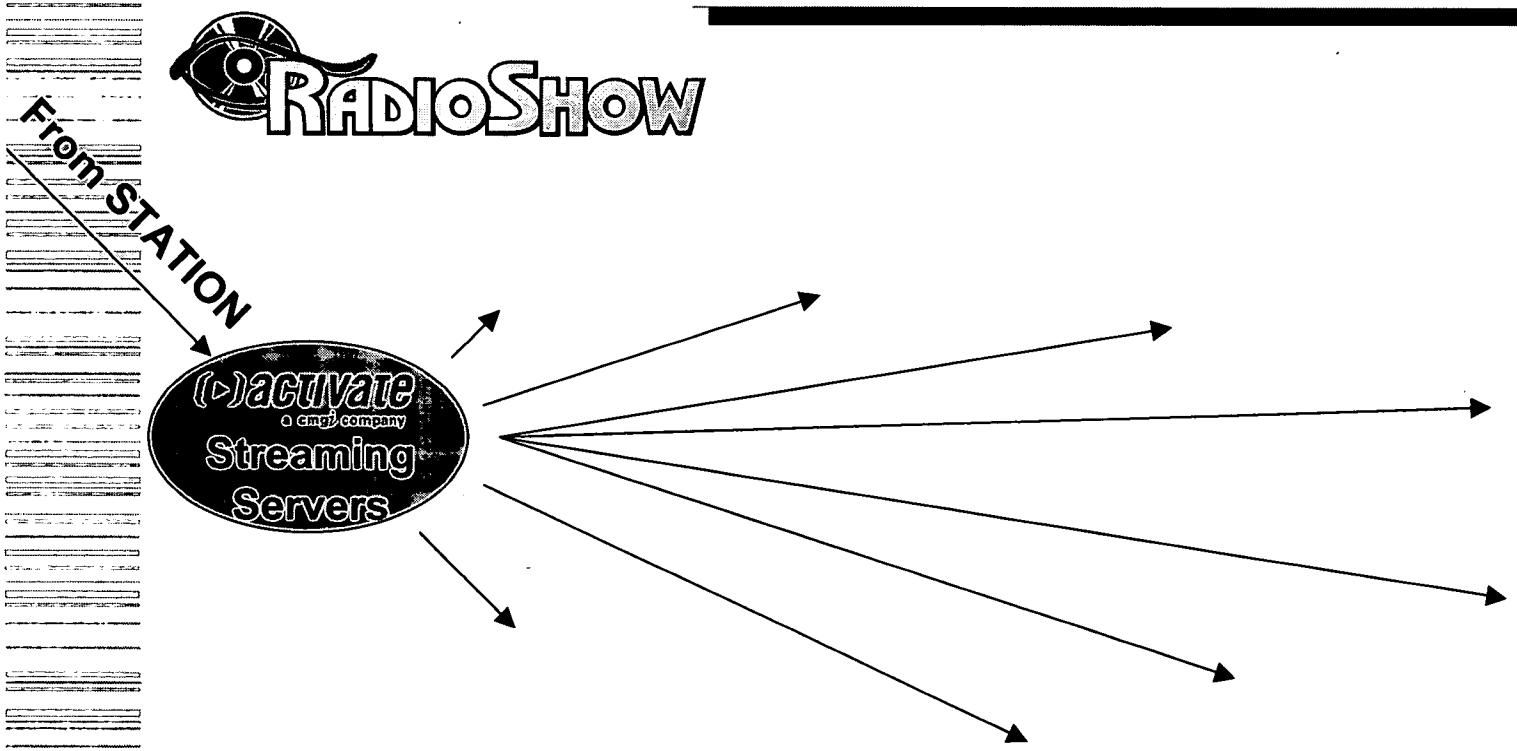
But…During spot breaks the RCS *SplitStream* system kicks in



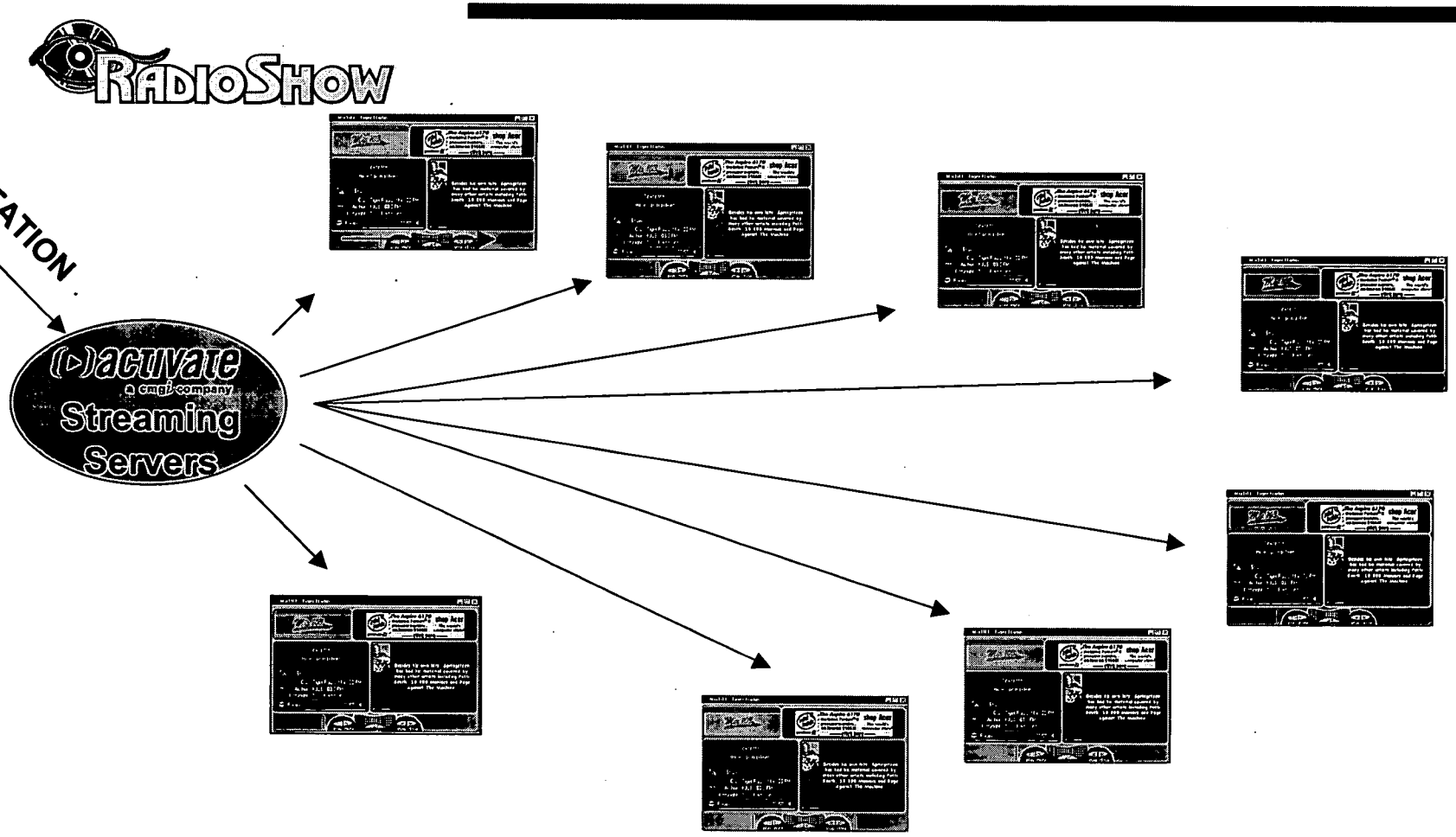
SplitStream sends different spots to targeted URLs.



When Spot Break is over, *RadioShow* resumes…



...Same program content to all, with continuously changing graphics



(▸)activate a cmgi company RCS

Using *SplitStream*

RadioShow

FROM STATION

activate a cmgi company

Streaming Servers

Ad Server

activate a cmgi company RCS

At the Station…

- Just add streaming encoder
- Easy software installation
- Analog or digital radio stations
- Works with all major digital automation systems

activate a cmgi company RCS

TOBTGO"OS/SS6GO

Ad Insertion Demo

Proprietary & Confidential

activate a cmgi company

RCS

Ad Insertion Functionality Highlights

- Post encoding solution
- Compatible with virtually all automation systems
- Compatible with non-automated stations
- Sell same avails to multiple customers = more revenue
- Scalable & redundant

activate a cmgi company RCS

Ad Insertion Functionality Highlights

- Dynamic campaign management at national, regional or local level
- Demographic and/or profile based targeting
- Rich, web-based reporting
- Leverages existing relationships with advertisers and agencies

Proprietary & Confidential

Pricing Model

- CPM Model Includes:
  - Ad insertion software
  - Provisioning, placement and ad serving systems
  - Streaming of audio ads
  - Detailed reporting
  - 24/7 Customer service

(▸)activate a cmgi company RCS

Additional Services

- Extended hosting of ad spots
- Encoding of ad spots
- Customized reporting
- Installation
- Outsource trafficking

activate a cmgi company RCS

Radio Show Functionality Highlights

- E-commerce links to purchase CD's
- Compatible with virtually every automation system
- Compatible with non-automated stations
- Rich administration tools to customize info displayed to consumer

Proprietary & Confidential

Radio Show Functionality Highlights

- Station branding synchronized with stream
- Graphics/banners synchronized with stream
- Title and artist on primary screen

Proprietary & Confidential

Pricing Model

- Software is leased
- Licensed annually
- Cost based on the features you choose

activate a cmgi company RCS

Getting Started

- 2 PC's (one dedicated, one non-dedicated)
- Installation of ad serving software
- Installation of RCS software
- Frame relay/router (suggested)
- Streaming with Activate
- Internal resource to manage ad operations

(▸)activate a cmgi company RCS

Summary

- Dynamic, live ad insertion
- End-to-end management system
- Proven, experienced team
- Minimal disruption to station activities
- Now Playing and Ad Insertion are seamlessly integrated

(▸)activate a cmgi company RCS

TOBTSD.055569

APPENDIX B

Dynamic Ad Insertion

Technical Specification

Version 0.2

September 17, 2001

This document contains trade secrets and proprietary information of ACTIVATE.NET. No use or disclosure of the information contained herein is permitted without prior written consent.

Activate.net
1130 Rainier Avenue, South
Seattle, WA 98144
(206) 830-5300
http://www.activate.com

TABLE OF CONTENTS

1.1 SCENARIOS 1

1.1.1 Introduction *1*

1.1.2 Scenario: Connect (Two embedded players) *2*

1.1.3 Scenario: Connect (Single embedded player) *2*

1.1.4 Discussion: Differences between connect scenarios *3*

1.1.5 Ad break (Two embedded players) *3*

1.1.6 Ad break (Single embedded player) *5*

1.1.7 Features omitted from the sequence diagrams *5*

1.2 TECHNICAL SPECIFICATIONS NOT CAPTURED IN THE SEQUENCE DIAGRAMS 5

1.2.1 ASX event cycling *5*

1.2.2 ASX file request *7*

1.2.3 Parameters passed for ad request *7*

1.3 SEQUENCE DIAGRAMS 8

1.3.1 Connect (two embedded players) *8*

1.3.2 Connect (single embedded player) *9*

1.3.3 Buffer break (two embedded players) *10*

1.3.4 Buffer break (single embedded player) *11*

1.3.5 Ad break (two embedded players) *12*

1.3.6 Ad break (single embedded player) *13*

Exhibit 2150

1.1 Scenarios

1.1.1 Introduction

The main sequence of events that is really interesting for ad insertion is the one that leads to the individualized ad being inserted into the radio stream. However, the correct environment has to be set up to facilitate the success of this sequence when the ad break happens. This environment is established when the listener connects to the radio station. Thus we also have to capture the sequence of events that establishes the correct environment when the player connects to an ad-enabled radio broadcast.

Moreover, we have three different scenarios (single embedded player, two embedded players and standalone player) that behave slightly different, both during initialization as well as during the ad break. Which of these three scenarios occurs depends on how the listener connects to the radio broadcast stream and potentially on the operating system on which the player runs.

Link type	Link to web page at radio station		To stream directly[1]
OS	Win 95, 98, 2000	Win NT 4	Any supported
Scenario	Two embedded players	Single embedded player	Stand alone player (not yet supported)

The preferred scenario is "Two embedded players", however Windows NT 4 does not support two players to simultaneously control the audio output. Thus a single embedded player is used on that platform resulting in potentially a bit more "dead air" at the end of the inserted ad before resuming the regular radio broadcast.

The following table summarizes the capabilities in the different scenarios:

	Two embedded players	Single embedded player	Stand alone player
"Dead air"	Minimal	5 sec	?
Synchronized banner ads	Yes	Yes	No (when avail.)
Individually targeted ads	Yes	Yes	No (when avail.)

[1] E.g. at Microsoft Media web site

1.1.2 Scenario: Connect (Two embedded players)

Step 1: The user opens the main web page of the radio station (e.g. www.kosu.org/main.html)

Step 2: The user click on a link in order to listen to the live broadcast

Step 3: In response to the click the browser opens a new window and instructs it to load a page with the embedded player(s) (e.g. www.kosu.org/live.html)

Step 4: The browser retrieves the page from the radio station's web server

Step 5: In order to serve the request the server determines the operating system of the machine that requests the page and tests whether it is Windows NT. *This scenario assumes that the OS is **not** NT.*

Step 6: Based on the result of step 5 the server ensures that the "two player page" is returned to the browser.

Steps 7 - 12: As part of loading the web page the browser executes a script.

Step 7: The script launches an instance of the Windows Media player that appears visually embedded into the web page.

Step 8: The script instructs the player to load the play-list ("ASX file") from the Activate web server. A query string indicates that the version for the embedded player is requested.

Step 9: The player retrieves the play-list.

Step 10: The player starts retrieving the stream as specified by the play-list.

Step 11: The script launches a second instance of the Windows Media player, in this case invisible to the user.

Step 12: The script instructs the first player to start playing

1.1.3 Scenario: Connect (Single embedded player)

Step 1: The user opens the main web page of the radio station (e.g. www.kosu.org/main.html)

Step 2: The user click on a link in order to listen to the live broadcast

Step 3: In response to the click the browser opens a new window and instructs it to load a page with the embedded player(s) (e.g. www.kosu.org/live.html)

Step 4: The browser retrieves the page from the radio station's web server

Step 5: In order to serve the request the server determines the operating system of the machine that requests the page and tests whether it is Windows NT. *This scenario assumes that the OS is NT.*

Step 6: Based on the result of step 5 the server ensures that the "single player page" is returned to the browser.

Steps 7 - 11: As part of loading the web page the browser executes a script[2].

Step 7: The script launches an instance of the Windows Media player that appears visually embedded into the web page.

Step 8: The script instructs the player to load the play-list ("ASX file") from the Activate web server. A query string indicates that the version for the embedded player is requested.

Step 9: The player retrieves the play-list.

Step 10: The player starts retrieving the stream as specified by the play-list.

Step 11: The script instructs the first player to start playing

1.1.4 Discussion: Differences between connect scenarios

The main difference between the two- and single player scenarios lies in the scripts that are control the embedded player(s). In one case the work is accomplished with two players (resulting in minimal "dead air") while in the other there is only one player to work with.

1.1.5 Buffer break (Two embedded players)

Step 1: Based on the parsed log-files the SplitStream software periodically inserts an "OpenEvent" into the stream when an ad break is anticipated but **before** it actually occurs.

Step 2: The inserted event is transmitted with the encoded broadcast signal to the Windows Media server at Activate.

Step 3: The inserted event is transmitted with the encoded broadcast signal from the Windows Media server to the Windows Media Player at the listener's PC.

Step 4: The Windows Media player looks up the appropriate action for the inserted event in the ASX file.

Step 5: As specified by the ASX file for this event the Windows Media player invokes a script in the web page into which the player is embedded.

Step 6: The script in the page combines information from the event with demographic information (either maintained in the page and/or e.g. persisted via a cookie) to assemble the add request from Ad Manager (running at Activate)

---

[2] The script contained on the "two player" and "single player" pages is significantly different.

[3] The address of this page is presumably stored in a database of radio stations.

[4] The address of the play-list is presumably stored in a database of radio stations

Step 7: The script requests the ad from Ad Manager which returns an URL to an ASP file that actually contains ASX content. (instead of a bitmap as it does for banner ads).

Step 8: The script instructs the second player to pre-buffer the stream specified by the ASP file returned from Ad Manager

Step 9: Windows Media player uses the ASX provided in step 8 to connect to a Windows Media server (which may or may not be the same as the one streaming the radio station) and to start buffering the ad content.

NOTE: The ad does not start playing at this point in time.

1.1.6 Buffer break (Single embedded player)

Steps 1-7: Identical to the scenario in the previous section.

Step 8: The script instructs the player to pre-buffer the stream specified by the ASP file returned from Ad Manager

Step 9: Windows Media player uses the ASX provided in step 8 to connect to a Windows Media server (which may or may not be the same as the one streaming the radio station) and to start buffering the ad content.

NOTE: The ad does not start playing at this point in time.

1.1.7 Ad break (Two embedded players)

Step 1: The DJ (either live person or simulated by a computer program) indicates the upcoming ad break to the RCS SplitStream software. ("Contact closure")

Step 2: The RCS SplitStream software inserts a Windows Media event into the live stream being encoded at the radio station.

Step 3: The inserted event is transmitted with the encoded broadcast signal to the Windows Media server at Activate.

Step 4: The inserted event is transmitted with the encoded broadcast signal from the Windows Media server to the Windows Media Player at the listener's PC.

Step 5: The Windows Media player looks up the appropriate action for the inserted event in the ASX file.

Step 6: As specified by the ASX file for this event the Windows Media player invokes a script in the web page into which the player is embedded.

Step 7: The script mutes the first player (with the radio broadcast signal)

Step 8: The script set the volume of the second player (with the as signal) to normal and instructs it to play the buffered ad..

DRAFT - 05/05/2001

1.1.8 Ad break (Single embedded player)

Step 1: The DJ (either live person or simulated by a computer program) indicates the upcoming ad break to the RCS SplitStream software. ("Contact closure")

Step 2: The RCS SplitStream software inserts a Windows Media event into the live stream being encoded at the radio station.

Step 3: The inserted event is transmitted with the encoded broadcast signal to the Windows Media server at Activate.

Step 4: The inserted event is transmitted with the encoded broadcast signal from the Windows Media server to the Windows Media Player at the listener's PC.

Step 5: The Windows Media player looks up the appropriate action for the inserted event in the ASX file.

Step 6: As specified by the ASX file for this event the Windows Media player invokes a script in the web page into which the player is embedded.

Step 7: The script in the page instructs the player to switch to playing the pre-buffered stream.

1.1.9 Features omitted from the sequence diagrams

- Synchronized banner ads
- Can we use commands embedded in the ASF file for the ad to trigger the synchronized banner ad? (E.g. pass the URL to the banner ad as the parameter of the event)

1.2 Technical specifications not captured in the sequence diagrams

1.2.1 ASX event cycling

1.2.1.1 Introduction

The RCS inserts an event into the stream to indicate an ad break to the player (step 2 of all ad break scenarios). This section specifies the algorithm for choosing the appropriate event.

The requirements for this are

(i) The length of all ads added together must match the length of the ad break in the program.

(ii) Within the first constraint it should be possible adjust the mix ad lengths to the available inventory of ads.

Example: Assume a radio station has only 2-minute ad breaks and available ads for that station are 25% 30-second ads, 50% 60-second ads and 25% 90-second ads. The desire is to use ads up evenly.

To do this we need 2 different events: One that combines two 60-second ads into one 2-minute break and one that combines a 30-second ad with a 90-second ad. Using these two events in a 1-to-1 ratio will achieve the goal of using up the ads evenly.

The example covered only an almost trivial case. In general, a radio station will have ads breaks of different lengths and the mix of available ads will be much more complicated. Moreover, both aspects (the lengths of the ad breaks at the station and the mix of available ads) will change over time. Thus a mechanism must be available to address this issue. Otherwise, we run the risk of being able to serve the ads we sold quickly enough (because they are of the wrong length) or to use up ads of a certain length too quickly.

Unfortunately we cannot solve the problem by requiring that mix of requested ads is changed at the RCS software that inserts the events into the stream because this would require SplitStream were aware of information such as the inventory of ads which is kept at the engage server. It is clearly undesirable to architect the system such that SplitStream needs a connection to Ad Manager.

The implication is that SplitStream knows only about the length of the ad break in the radio broadcast and can choose for each length only from a pre-determined set of events. However, what can change is the *interpretation* of the event. In other words we need a system in which the ASX file can be adjusted in such a way that the right mix of ad lengths is achieved.

0050545.0050404

1.2.1.2 Design

Assumptions:

- Ads can be 30, 60, 90, 120, 150 and 180 seconds long.
- Ad breaks can be between 30 seconds and 5 minutes long, in 30-second increments.
- The order of ad lengths is irrelevant, i.e. a 30-30-60, a 30-60-30 and a 60-30-30 break are equivalent. There is no reason to specifically request one or the other.

Event names:

For each length `l` (in seconds) of an ad break use `n` (I suggest `n` being initially 25) events of the format "`<<l>>s-break-<<i>>`". Where `i` ranges from 1 to `n`. I.e for 1-minute breaks we'd use the events `60s-break-1, 60s-break-2, 60s-break-3,..... 60s-break-25`. The SplitStream software will simply cycle through event names. Ideally the number n (the maximum number after which the counter goes back to one) is not hard coded but dynamically determined, e.g. read from an .ini file or the registry. The ASX files will be adjusted to achieve a mixture of ads that will exhaust the inventory as desired. E.g. `60s-break-1` through `60s-break-16` would map to two 30-second ads, the remaining events to one 60-second ad.

1.2.2 ASX file request

We need to agree and document a mechanism to distinguish requests for ASX files from stand-alone players and embedded players. The sequence diagram suggests using query strings. An alternative would be to use different URLs. Preferences? Opinions? Other alternatives?

1.2.3 Parameters passed for ad request

When the ad is requested from Ad Manger several parameters can be passed along with the request in the form of query strings. These parameters allow Ad Manager to target the ad more precisely.

The format of the query strings is determined by Ad Manager but is quite flexible and extensible.

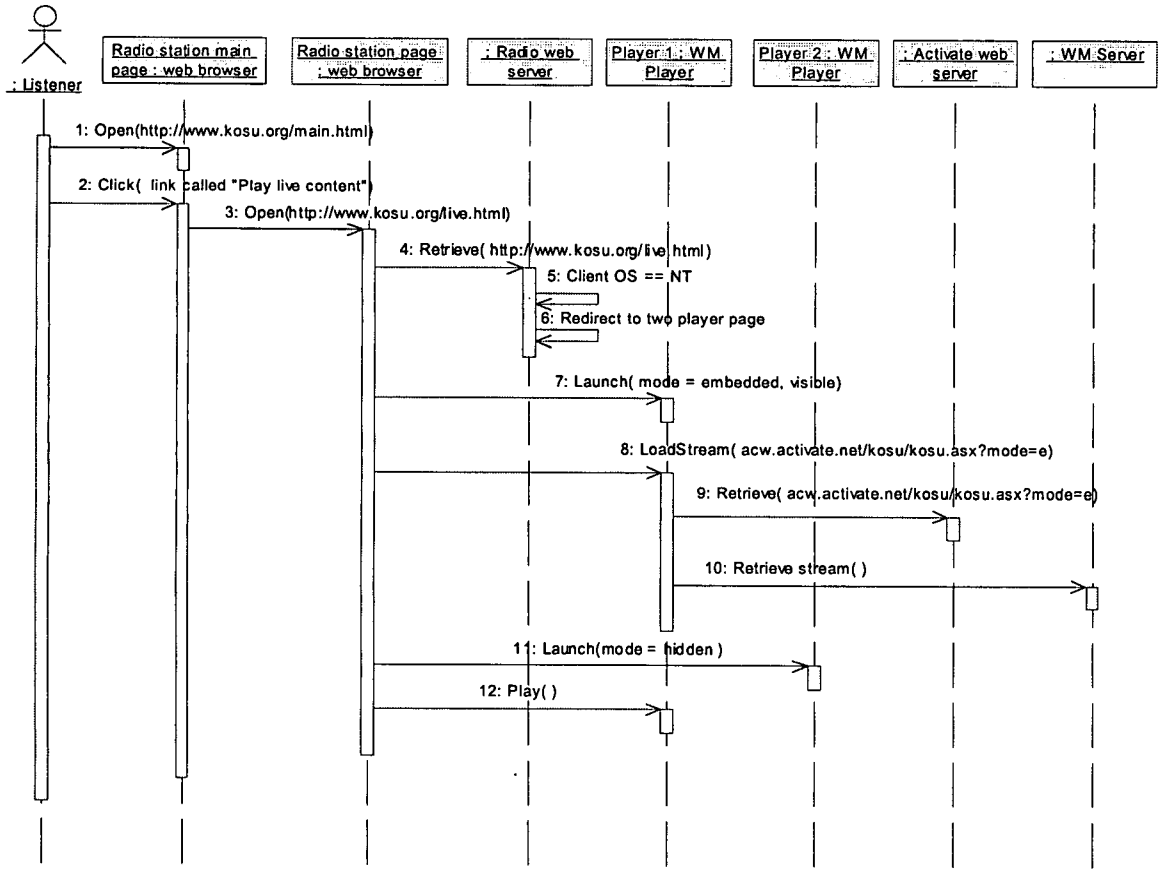
The first step to determine which parameters can be sent is to figure out what information is available. In the case of the stand-alone player the whole request to the ad manager must be contained in the ASX file. The ASX file is shared between all listeners of that radio station. In other words, the information transmitted cannot be more specific than the particular radio station but could include broader categories such as the genre typically played by that radio station or the stations geographical location.

In the case of the embedded player, the script in the HTML page controlling the player can add additional information. However, since we're not planning to force the user to register before accessing the radio stream, I don't see that this additional information could be anything beyond a cookie that is used by Ad Manager to track the identity of repeat visitors.

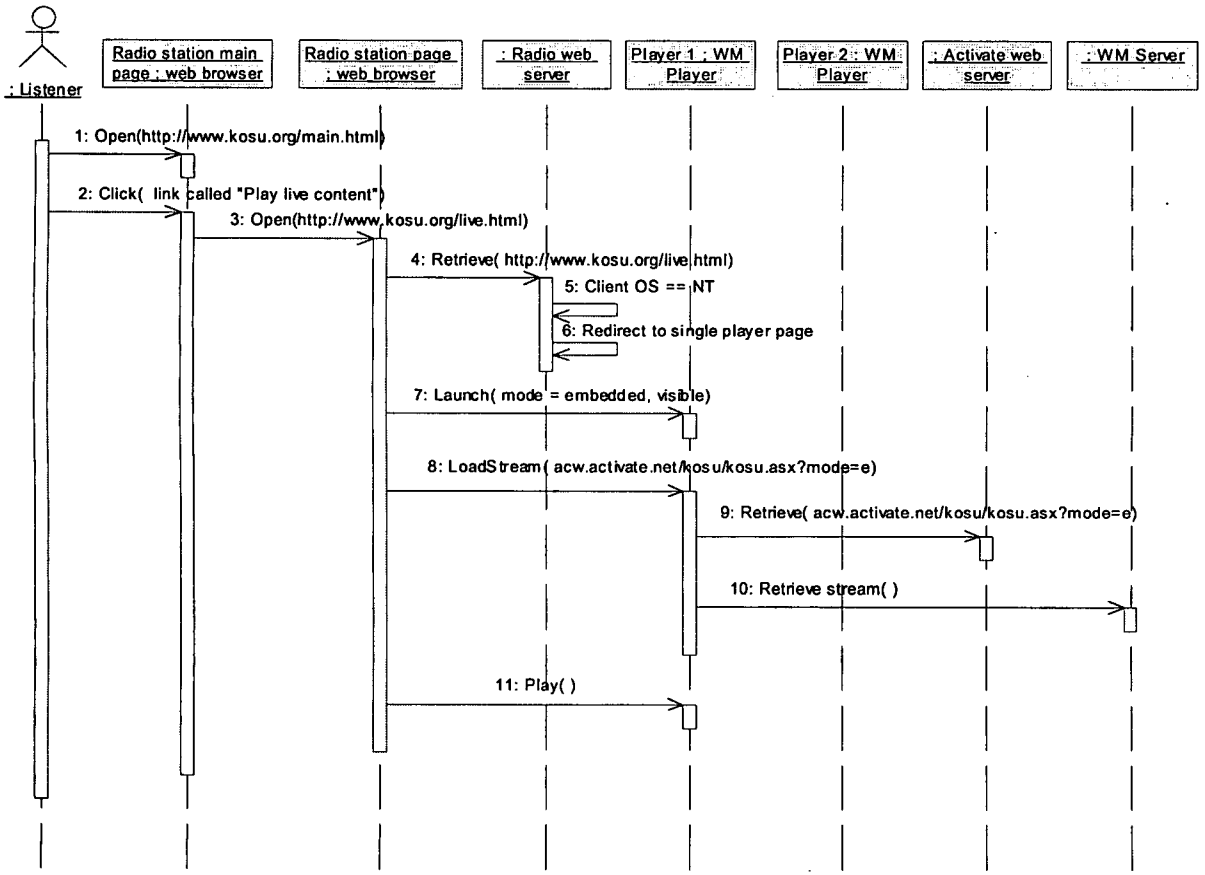
0055450.0074

1.3 Sequence diagrams

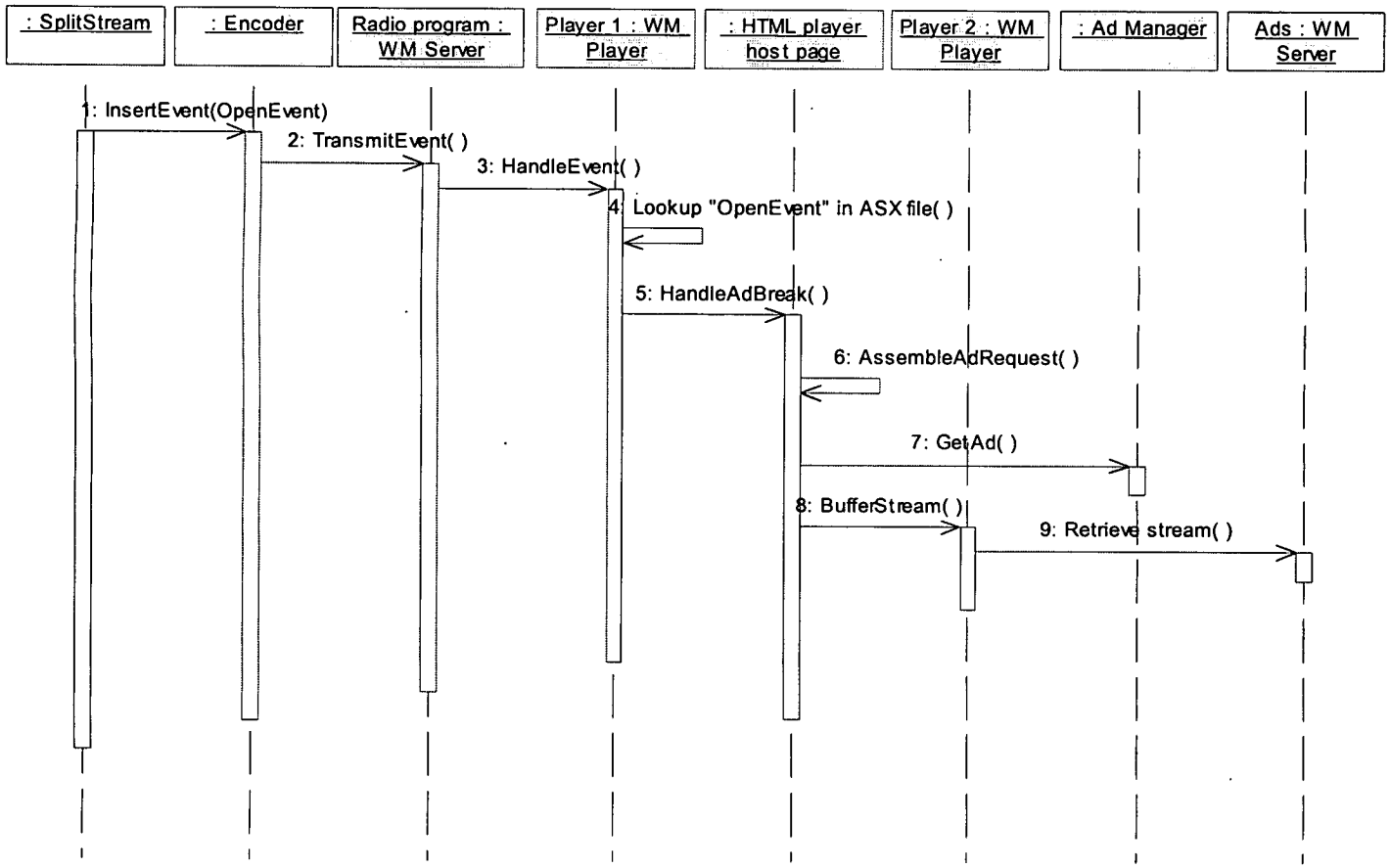
1.3.1 Connect (two embedded players)



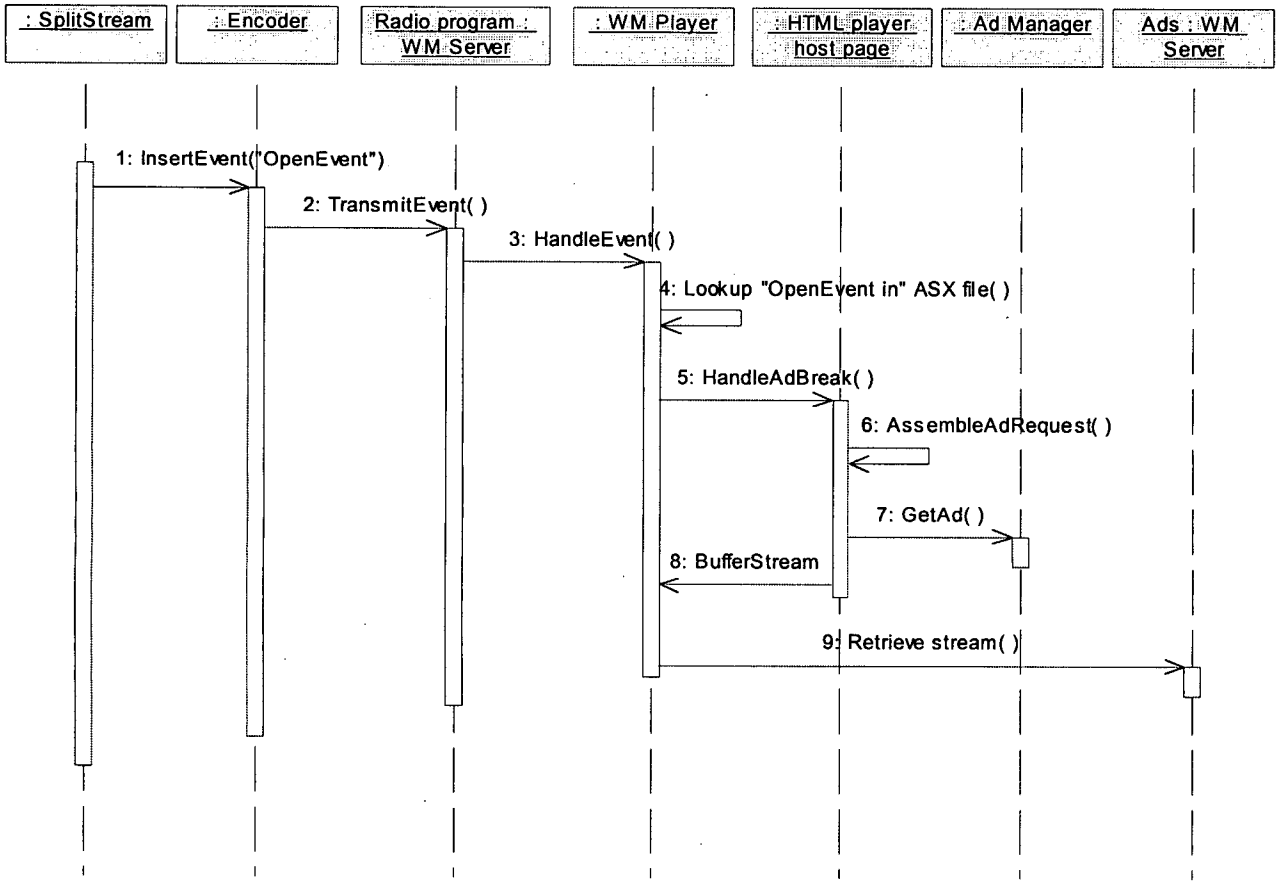
1.3.2 Connect (single embedded player)



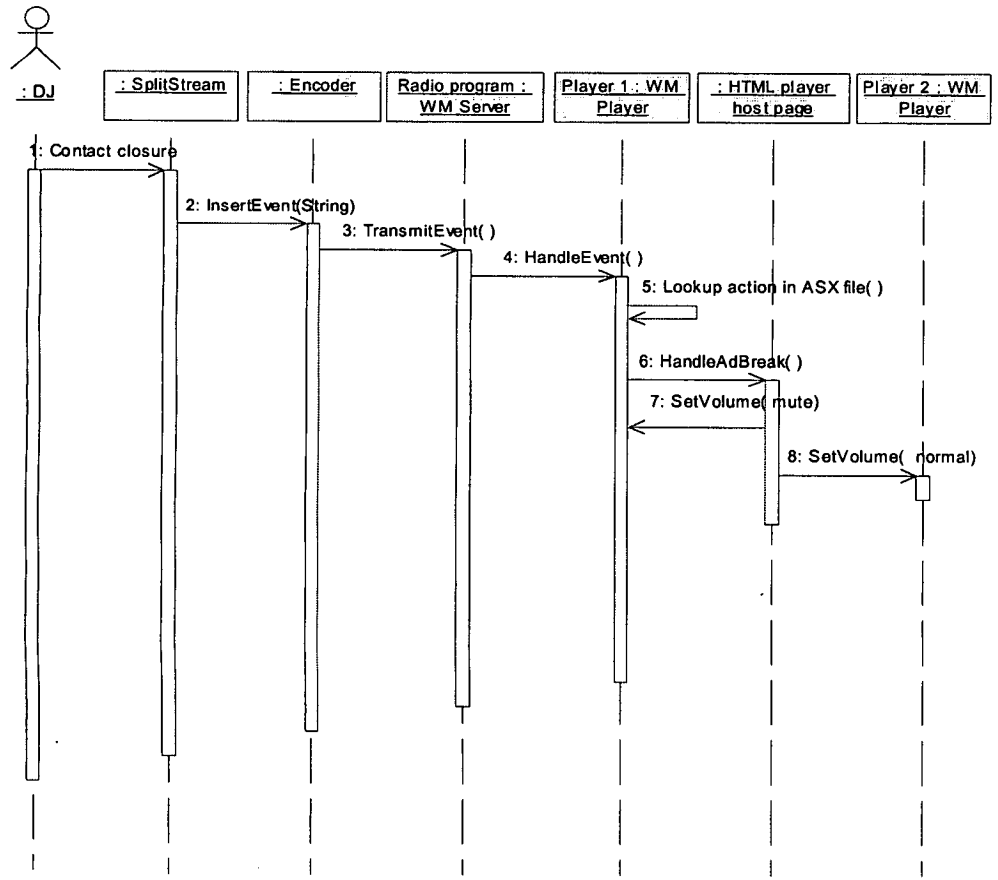
1.3.3 Buffer break (two embedded players)



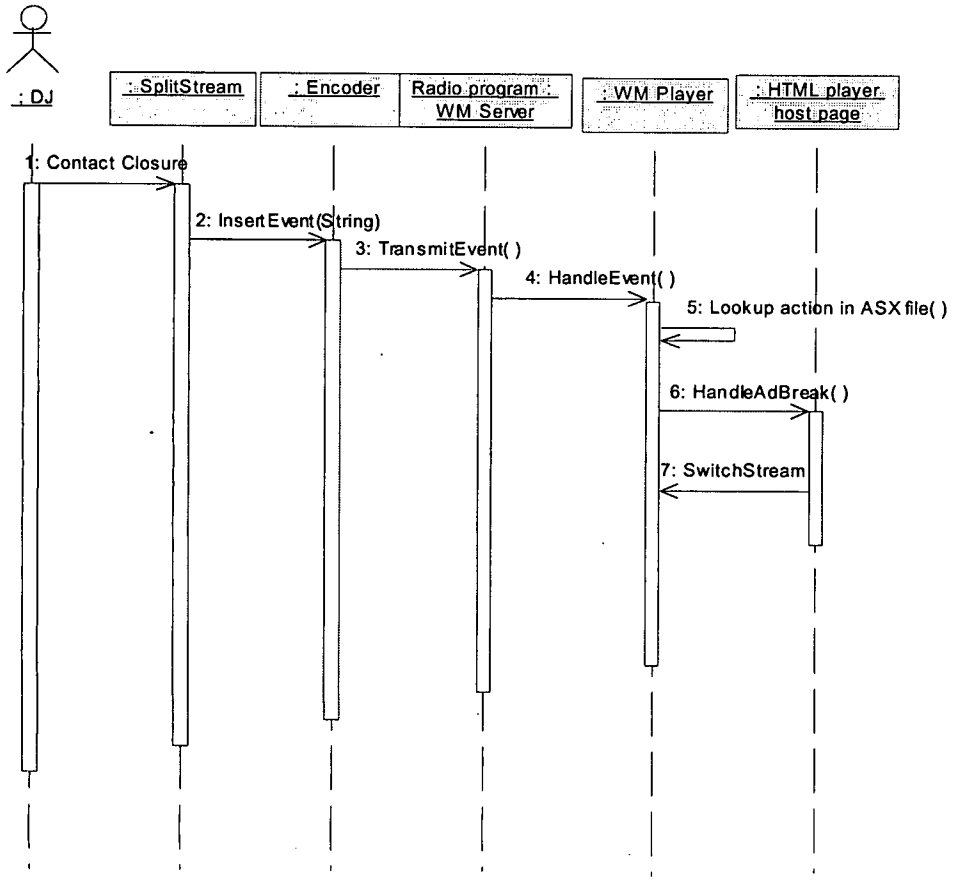
1.3.4 Buffer break (single embedded player)



1.3.5 Ad break (two embedded players)



1.3.6 Ad break (single embedded player)



APPENDIX
C

T0815D-055565D

Station side break

```
<%@ LANGUAGE="JavaScript" %>
<%
//Copyright 2000  All rights reserved.
   Response.Expires = 0;
   Response.ContentType = "video/x-ms-asf";

   var theURL =
"http://64.85.69.243:98/xtserver/site=siteone/area=rcs/aamsz=";
   var adDuration = Request.QueryString("adDuration"); // in seconds
   if (adDuration == "" || isNaN(adDuration))
      theURL += "30";
   else
      theURL += adDuration;
   theURL += "_SEC_AUDIO_16K";
   //var theURL = "rcsBreak30.asp";

%><ASX version = "3.0">
<ENTRYREF HREF="<%= theURL %>" CLIENTBIND="NO" />
</ASX>
```

Ad request from Player

Response Header
HTTP/1.0 200 OK
Server: Accipiter Direct AdServer/5.1.0.2.5 for NT (Pentium)
Content-Type: video/x-ms-asf
Content-Length: 582
Content – Profile: 120,4,30,30,30,30,news,98xxx,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
Date: Fri, 13 Oct 2000 19:20:51 GMT
Pragma: no-cache
Cache-control: no-cache
Set-Cookie: StationId
GUID=FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF;
expires=Sunday, 29-Feb-2004 23:59:59 GMT; path=/; domain=64.85.69.243;

Return from the Adserver

```
<ASX version = "3.0">
<ENTRY CLIENTSKIP="NO">
    <REF HREF="mms://myserver/myCommercial.asf" />
    <MOREINFO HREF="http://myserver/myCommercial.gif"
TARGET="_blank" />
</ENTRY>
</ASX>
```

Ad Server Scheme

Proposed Schema

1) SITE = Station ID (call letters)
example = KPIG; KPLU; etc.

2) SIZE = (Ad type + length)_bitrate
example = 30secaudio_56K
additional values TBD

3) AREA = Station format (genre)
example = rock
additional values TBD
SUB-AREA 1 = Station geography
example = rock.seattle; classical.california
additional values TBD

Capturing Other Data Elements

1) Time of Day - in ASX
2) Calendar Date - in ASX
3) Geographic Area (City, State, Zip Code, MSA, DMA, Country, TimeZone, AreaCode) - in Ad Manager GeoTargeting module
4) Arbitron Demographics (Age, Gender, Other) - Declared data not tracked by Engage Ad Manager or Profiling
5) Operating System - Target criteria via Ad Manager User Interface
6) Browser Type - Target criteria via Ad Manager User Interface
7) Player Version/Type - tracked before AdManager
8) Station Group Affiliation - Target criteria via Ad Manager User Interface
9) Sales Rep ID - captured in Campaign Data entered through UI
10) Advertiser ID - captured in Campaign Data entered through UI
11) Campaign Name - captured in Campaign Data entered through UI
12) Advertisement Type (e.g. 'Audio+Banner') - set up as custom format in UI
13) Impression Type (Paid, Bonus, PSA, Station Promo) - captured in Campaign Data entered through UI
14) Advertisement Cost (CPM, Clicks) - captured in Campaign Data entered through UI
15) Ad Breaks (Impressions) - captured in Campaign Data entered through UI
16) Ad Avails - tracked in Inventory Projection reports
17) Stream Serving Location - custom tag within Sub-Area

EXHIBIT 10.54

This Page is Inserted by IFW Indexing and Scanning Operations and is not part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☒ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☐ FADED TEXT OR DRAWING
- ☐ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☒ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☐ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: _____

IMAGES ARE BEST AVAILABLE COPY.
As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.